Docket No.: 30320/17230
P17230

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
APPLICATION FOR UNITED STATES LETTERS PATENT

Title:

A SYSTEM AND METHOD FOR DETERMINING PROCESSOR UTILIZATION

Raju Yasala

141 Saratoga Ave., Apt. 1227
Santa Clara, California 95051

I hereby certify that this correspondence is being deposited with the U.S. Postal Service as Express Mail, Airbill No. EV323775435US, in an envelope addressed to: MS Patent Application, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on the date shown below.

Dated: November 12, 2003 Signature: (Richard Zimmermann)

A SYSTEM AND METHOD FOR DETERMINING PROCESSOR UTILIZATION

Field Of The Invention

[0001] This invention relates generally to using processor current consumption data to determine processor utilization, and more specifically, to a system and method for determining processor utilization without compromising performance of the processor.

Background Of The Invention

[0002] A computer processor has a finite number of instructions it can process per unit of time. Complex processor instructions require more processor resources per unit of time as compared to simple processor instructions. When the computer processor reaches its limit of instruction processing capability, software applications running on that processor take longer to complete. The amount of processor resources being used by the processor is known as processor utilization and is usually displayed as a percentage (e.g. 50% utilization).

[0003] In the past, determining processor utilization was accomplished by a utilization software program (utilization application) running on the same processor of which utilization information was desired. This arrangement had two significant drawbacks. First, the utilization application added additional overhead to the processor and resulted in the processor's performance being compromised. Running the utilization application on the same processor for which utilization information is desired results in an inability to differentiate the amount of resources consumed by the utilization application versus other applications running on the processor.

[0004] Second, the utilization information provided by the utilization application reflected a time value in which software applications were being executed by the processor. This value was indicative of how much time a processor spent on a software application rather than an indication of the amount of processor resources being used by the processor. As a result, a high value for processor utilization was not necessarily indicative of the amount of resources the processor was using, but merely indicated a software application consuming a large part of the processor's time.

Brief Description Of The Drawings

[0005] FIG. 1 is a block diagram of a system in accordance with the invention.

[0006] FIG. 2 is a plot of percent processor utilization versus time for various revisions of a software application executed by the target processor, in accordance with one embodiment of the invention.

[0007] FIG. 3 is a flowchart illustrating steps of obtaining data indicative of processor utilization from the target processor, in accordance with the invention.

Detailed Description Of The Invention

[0008] A system 100 for determining processor utilization in a target computer system 105 is illustrated in Figure 1. The target computer system 105 has a memory 110 and a target computer processor 115 (hereinafter target processor) being supplied power from a power supply 120 through a power line 125. The system 100 has a memory 130, a processor 135, and an input 140.

[0009] In accordance with the invention, a current sensor 145, such as a hall-effect sensor, measures real-time current consumed by the target processor 115. The real-time current consumed by the target processor 115 is proportional to the utilization of the target processor 115. The real-time current is compared to a maximum current value, the maximum current value indicative of the current consumed by the target processor 115 when fully utilized, to determine the real-time percent utilization of the target processor 115. Control and/or communication between the system 100 and the target system 105 is accomplished with a control line 150.

[0010] A software application known as a utilization application controls the system 100 during data acquisition of the target system's processor 115. The utilization application is stored in the system memory 130 and executed by the system processor 135. Control of the system 100 includes collecting data from the current sensor 145, logging sensor data received at an input 140 to the system memory 130, controlling the number of samples collected per unit of time, calculating utilization data based on both the maximum current value and the data collected from the current sensor 145, and generating plots.

[0011] Calculation of processor utilization requires two pieces of data. The first piece of data is the real-time current being consumed by the target processor 115. The second piece of data is the current that the target processor 115 uses when it is 100% utilized.

[0012] The utilization application divides the value for real-time current used by the target processor 115 at any one moment by the value for maximum current that could be used by the target processor 115 to yield a percentage of processor utilization. The maximum current consumed by the target processor 115 is data that may be supplied by the target processor's manufacturer. However, for circumstances where the target processor's maximum current consumption is not known, a maximum current utility (software application) is used to cause the target processor 115 to utilize a maximum amount of the target processor's resources. The maximum current utility (MCU) is stored in the target system memory 110 and executed by the target processor 115 after being initiated by the utilization application via the control line 150. During execution of the MCU for a pre-determined amount of time, the current sensor 145 measures the current used by the target processor 115 and sends that current data to the system input 140 where the utilization application stores it to the memory 130 for later use in the processor utilization calculation. An average maximum current is calculated by the utilization application and is considered a "benchmark" of the target processor 115 for which all real-time activity is compared.

[0013] Data representative of average maximum current, determined by executing the MCU for a pre-determined amount of time at an earlier stage, is performed on the processor 115 at least once. This maximum current benchmark may be determined on an assembly line after manufacturing, or after the processor is installed in a computer system. For example, if the MCU determines the average maximum current to be 75 units, any subsequent current reading is divided by the maximum current value to compute the processor utilization. Thus, a real-time current measurement of 30 units would yield a processor utilization value of 40%.

[0014] Target system performance bottlenecks may be discovered and software architects will find processor utilization data useful during application tuning to verify performance effects among various software application versions, as shown in Figure 2. Data associated with program A.1 (version 1 of program A) 200 displays

significant processor utilization values at time units 4, 6, and 9. After the software architect modifies program A.1 (thereby creating version A.2) to utilize more processor resources at those times, the system 100 may be used to compare program A.1 200 with the optimized program A.2 210. The system 100 plots data for both programs A.1 200 and A.2 210 to allow the system architect to verify whether or not program A.2 210 is an improvement over the previous version. Figure 2 illustrates that the efforts of the software architect were successful such that optimized program A.2 210 has a higher processor utilization value at time units 4, 6, and 9 than previous version A.1 200. As such, target system performance bottlenecks may be reduced by using more processor resources.

[0015] Figure 3 illustrates steps of a method for determining processor utilization. Beginning at block 300, the host system 100 is configured. Configuration of the host system 100 includes connecting the current sensor 145 to the target processor 115 and adjusting user settings. One sample configuration collects processor utilization data and generates comparative plots, as shown in Figure 2, whereas other sample configurations may collect and plot processor utilization in real-time. At block 310 the MCU is installed on the target processor 115 for circumstances where a target processor's maximum current consumption is unknown. As discussed earlier, the MCU is a software application adapted to utilize a maximum amount of processor resources on the target processor. Measuring the current consumed by the processor when it is most utilized allows a benchmark to be established so that processor utilization is calculated as a ratio of the target processor's 115 maximum capabilities. Block 320 involves setting up a target software application on the target system 105 wherein the software application's demand on the target processor 115 is unknown. More than one target software application may be installed/configured on the target system 105 to ascertain how utilized the target processor 115 is during the execution of those software applications. The target application is started for a pre-determined amount of time at block 330 and the effects of that application on the target processor 115 are collected by the current sensor 145 and logged at block 340. After the target software application has stopped executing, the MCU is started at block 350 and the effects of the MCU on the target processor 115 are collected by the current sensor 145 and logged at block 360. Upon completion of data collection at block 360, an average maximum processor current value is calculated at block 370. The host system 100

now has enough data at block 380 to calculate the percentage of processor usage caused by executing the target application(s). The percent of processor utilization is obtained by dividing the current consumed by the target processor 115 during execution of the software application(s) by the average maximum processor current consumed when the MCU was executed. The host system 100 plots the percent processor usage versus time at block 390.

[0016] Although the foregoing text sets forth a detailed description of numerous different embodiments of a system and method for determining processor utilization, it should be understood that the legal scope of the method and apparatus is defined by the words of the claims set forth at the end of this patent. The detailed description is to be construed as exemplary only and does not describe every possible embodiment of the system and method for determining processor utilization because describing every possible embodiment would be impractical, if not impossible. Numerous alternative embodiments could be implemented, using either current technology or technology developed after the filing date of this patent, which would still fall within the scope of the claims defining the system and method for determining processor utilization.

[0017] Thus, many modifications and variations may be made in the techniques and structures described and illustrated herein without departing from the spirit and scope of the present invention. Accordingly, it should be understood that the methods and apparatuses described herein are illustrative only and are not limiting upon the scope of the system and method for determining processor utilization.